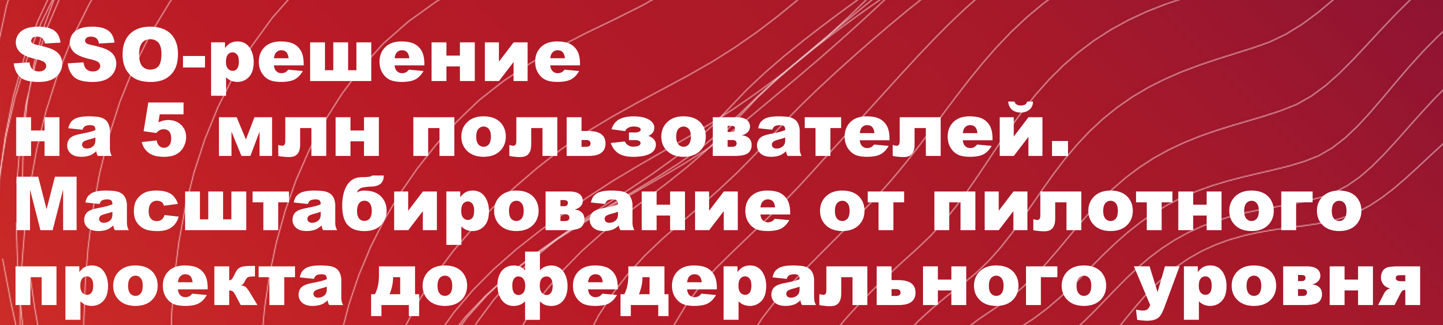


SSO-решение на 5 млн пользователей. Масштабирование от пилотного проекта до федерального уровня

Ирина Блажина
Николай Зайцев

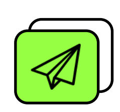






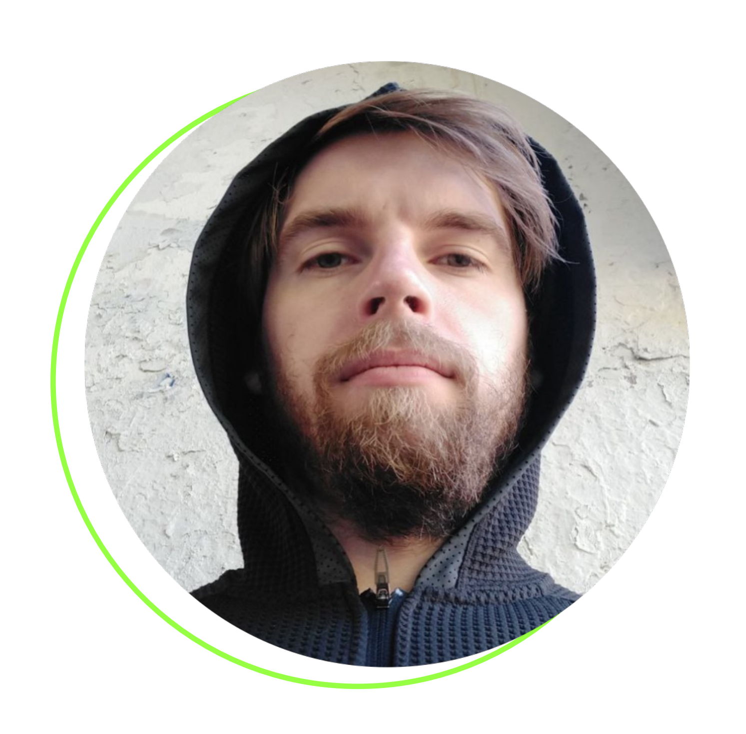
Ирина Блажина

Корпоративный архитектор X5 Tech



@A_Blair





Николай Зайцев

Архитектор решений X5 Tech



@gtjbtits

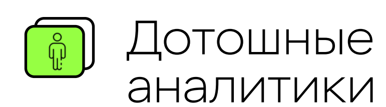
У НАС БОЛЬШАЯ И НАДЕЖНАЯ КОМАНДА X5ID

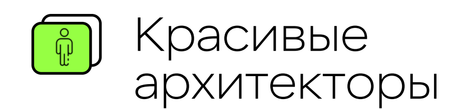




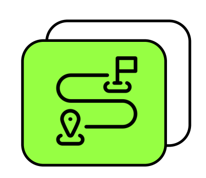
20+ человек

- Креативные разработчики
- Упорные DevOps
- Внимательные тестировщики









Предыстория появления гостевого SSO в X5



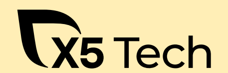
>20 тыс.

магазинов

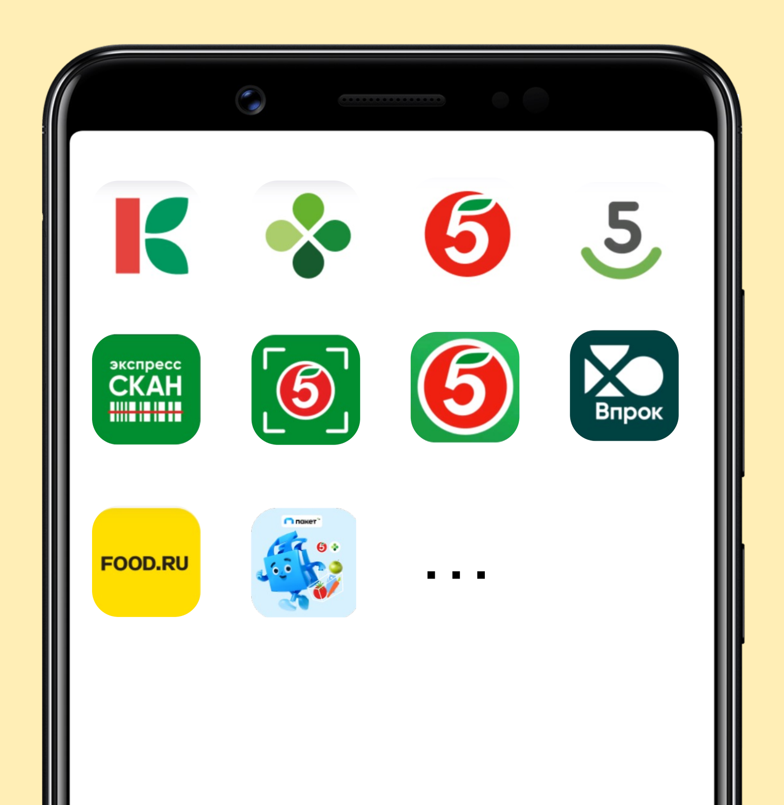
>15 млн

покупателей каждый день

У БОЛЬШОЙ КОМПАНИИ КЛИЕНТСКИХ СЕРВИСОВ ТОЖЕ МНОГО...



Приложения X5 Group







ЗАКОН КОНВЕЯ

«Организации проектируют системы, которые копируют структуру коммуникаций в этой организации»

ЗАКОН КОНВЕЯ НА ПРАКТИКЕ



ADEL@MAIL.RU
4631EVVA

ADEL@MAIL.RU
572122234

ADEL@MAIL.RU
9503JFWL
Впрок

ADEL@MAIL.RU
G21K43L3

ADEL@MAIL.RU
G21K43L3

ADEL@MAIL.RU
34547FDDH

ADEL@MAIL.RU
YTSDF467

ADEL@MAIL.RU
JDFGC4
FOOD.RU

ADEL@MAIL.RU
LOVE135
экспресс
СКАН

ADEL@MAIL.RU
7T83REDR

ИСХОДНЫЕ ДАННЫЕ

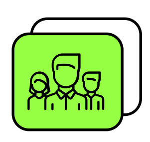




Крупная компания с большой клиентской аудиторией

ИСХОДНЫЕ ДАННЫЕ





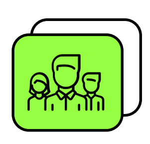
Крупная компания с большой клиентской аудиторией



Растущее число продуктовых сервисов

ИСХОДНЫЕ ДАННЫЕ





Крупная компания с большой клиентской аудиторией



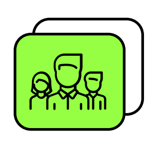
Растущее число продуктовых сервисов



Большие расходы на отправке sms для подтверждения входа

ИСХОДНЫЕ ДАННЫЕ





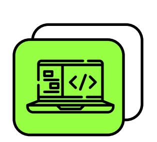
Крупная компания с большой клиентской аудиторией



Растущее число продуктовых сервисов

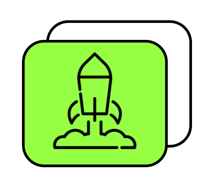


Большие расходы на отправке sms для подтверждения входа



Недовольные забывающие логины и пароли пользователи





СТАРТ ПРОЕКТА

ЧТО ПРЕДСТОИТ СДЕЛАТЬ



1

Реализовать единый вход (SSO)

для комфорта потребителей



Сделать быстро

за 2 месяца на пилот
на приложениях
с небольшой аудиторией



Сократить расходы в будущем

на локальных доработках и sms



Повысить безопасность

Обеспечить безопасный и надежный сервис для клиента

ОСОБЕННОСТИ ИНФРАСТРУКТУРЫ



Интернет

Клиент

ЦОД 1..К

DMZ

WAF

API GW

LB

Ресурс 1
(front-end)

...

Ресурс N
(front-end)

Ресурс 1
(back-end)

...

Ресурс N
(back-end)

Ресурс (API, веб-страница): 5ka.ru , perekrestok.ru ...

ОСОБЕННОСТИ ИНФРЫ



Интернет

Клиент

ЦОД 1..K

WAF

DMZ

API GW

Ресурс 1
(front-end)

Форма
входа 1

...

Ресурс N
(front-end)

Форма
входа N

LB

Ресурс 1
(back-end)

...

Ресурс N
(back-end)

Ресурс (API, веб-страница): 5ka.ru , perekrestok.ru ...

ОСОБЕННОСТИ ИНФРЫ



Интернет

Клиент

ЦОД 1..K

WAF

DMZ

API GW

Ресурс 1 (front-end)

Форма входа 1

...

Ресурс N (front-end)

Форма входа N

LB

Ресурс 1 (back-end)

Учетные данные 1

...

Ресурс N (back-end)

Учетные данные N

Ресурс (API, веб-страница): 5ka.ru , perekrestok.ru …





ПОИСК РЕШЕНИЯ

ПОЧЕМУ БЫ НЕ НАПИСАТЬ СВОЙ SSO - СЕРВЕР?





Небыстро

SSO, CORS, OIDC, OAuth, JWT



Сложно развивать

И нужно обеспечить поддержку разных языков и технологий



Потеря знаний

Если не по стандартам/уход разработчика - потеря знаний и времени

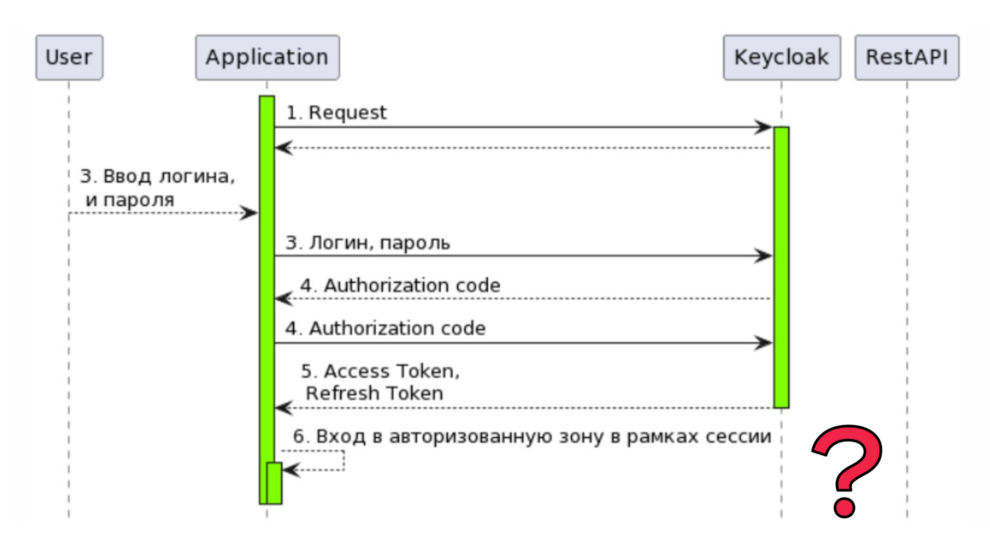


Что с безопасностью?

...

ВАРИАНТ РЕАЛИЗАЦИИ OAuth2 ИЗ ПРАКТИКИ







ВАРИАНТ РЕАЛИЗАЦИИ OAuth2 ИЗ ПРАКТИКИ



User
Application
Keycloak
RestAPI

1. Request

3. Ввод логина,
и пароля

3. Логин, пароль

4. Authorization code

4. Authorization code

5. Access Token,
Refresh Token

6. Вход в авторизованную зону в рамках сессии

7. Запрос API
без!!!
Access Token

8. OK

User
Application
Keycloak
RestAPI



Ученые Боннского университета провели исследование





Оценили навыки

создания систем безопасного хранения паролей у 43 программистов-фрилансеров





Большинство не задумывается о защите ключей

Если заранее не указать на необходимость защиты



Лишь 27% программистов

Использовали надежные алгоритмы шифрования, остальные отдали предпочтение слабым механизмам кодирования

ВЫБОР X5ID SSO



Готовое открытое решение

Возможность **доработок**

Работа **с БД**

Управление **сессиями**

Регистрация пользователей

Управление **пользователями и ролями**

OAuth/OIDC, JWT

SSO, CORS

Адаптеры для популярных языков



АРХИТЕКТУРА ПИЛОТА X5ID



АРХИТЕКТУРА ПИЛОТА X5ID



Интернет

Клиент

ЦОД

WAF

DMZ

API GW

ID.X5.RU
(front-end)

Форма
входа

Scan&Go
(front-end)

экспресс
СКАН

LB

Scan&Go
(back-end)

экспресс
СКАН

X5ID

X5ID SSO

KEYCLOAK

Учетные
данные

СХЕМА МАСШТАБИРОВАНИЯ — К ЧЕМУ ИДЕМ



РЕЗУЛЬТАТЫ ПЛАНИРОВАНИЯ





Выбранное решение

Должно удовлетворить все наши запросы



Архитектура пилота

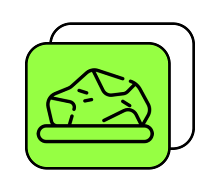
На небольшую аудиторию в 300 000 пользователей



План масштабирования

На 5 млн активных пользователей





ПОДВОДНЫЕ КАМНИ

ПРОБЛЕМА РОСТА ВРЕМЕНИ ПРОСТОЯ ПРИ ОБНОВЛЕНИИ



Больше пользователей → дольше обновление и восстановление после аварии.
Приемлемым считаем время простоя системы **не более трех минут**.

ПРОБЛЕМА РОСТА ВРЕМЕНИ ПРОСТОЯ ПРИ ОБНОВЛЕНИИ



Больше пользователей → дольше обновление и восстановление после аварии.
Приемлемым считаем время простоя системы **не более 3-х минут**.

ПРОБЛЕМА РОСТА ВРЕМЕНИ ПРОСТОЯ ПРИ ОБНОВЛЕНИИ



Больше пользователей → дольше обновление и восстановление после аварии.
Приемлемым считаем время простоя системы **не более 3-х минут**.

ПРОБЛЕМА РОСТА ВРЕМЕНИ ПРОСТОЯ ПРИ ОБНОВЛЕНИИ



Количество пользователей, тыс.

Больше пользователей → дольше обновление и восстановление после аварии.
Приемлемым считаем время простоя системы **не более 3-х минут**.

ПОВЕДЕНИЕ КЛАСТЕРА INFINISPAN



ОЖИДАНИЕ

ПОВЕДЕНИЕ КЛАСТЕРА INFINISPAN



ОЖИДАНИЕ

+

ПОВЕДЕНИЕ КЛАСТЕРА INFINISPAN



ОЖИДАНИЕ

JGROUPS

ПОВЕДЕНИЕ КЛАСТЕРА INFINISPAN



ОЖИДАНИЕ

JGROUPS

ПОВЕДЕНИЕ КЛАСТЕРА INFINISPAN



РЕАЛЬНОСТЬ

JGROUPS

ПОВЕДЕНИЕ КЛАСТЕРА INFINISPAN



РЕАЛЬНОСТЬ

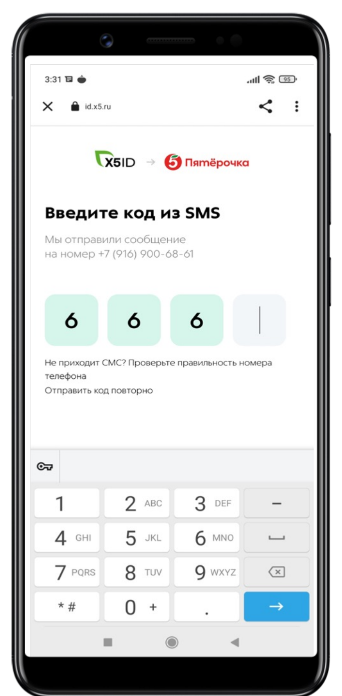
JGROUPS

JGROUPS

ПОЧЕМУ НАМ НЕЛЬЗЯ ТЕРЯТЬ СЕССИИ?

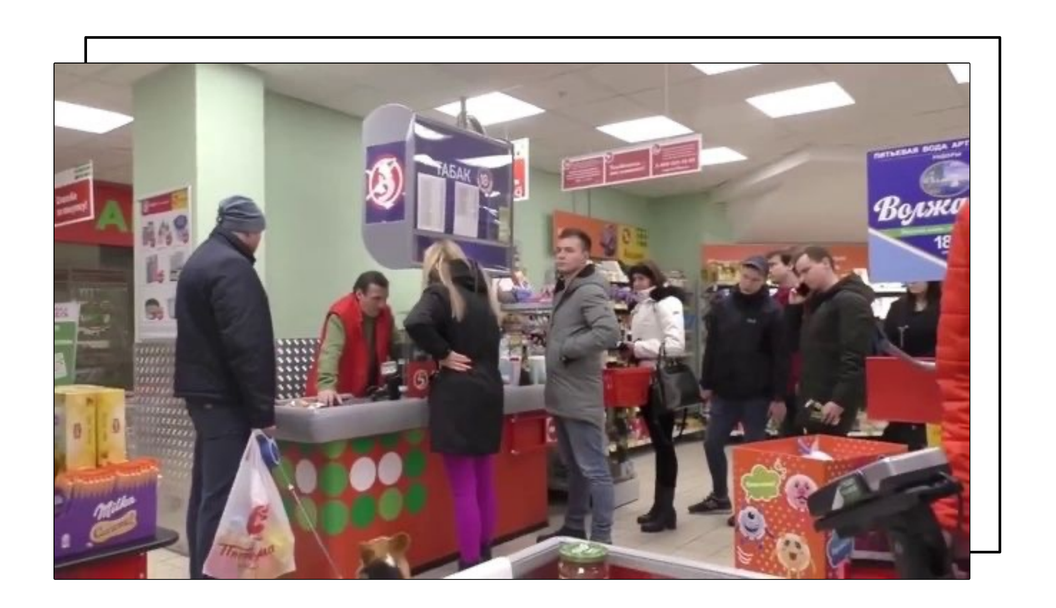
X5 Tech







Каждая авария или обновление с потерей сессий оборачивается для пользователей **разлогином на кассе**



ПРИЧИНА МЕДЛЕННОЙ ИНИЦИАЛИЗАЦИИ KEYCLOAK



LOADING...

INFINISPAN

POSTGRES

Каждый раз при загрузке происходит **прогрев кэша в Infinispan** из Postgres – перенос всех сессий из Postgres в кластер Infinispan

ПРИЧИНА МЕДЛЕННОЙ ИНИЦИАЛИЗАЦИИ KEYCLOAK

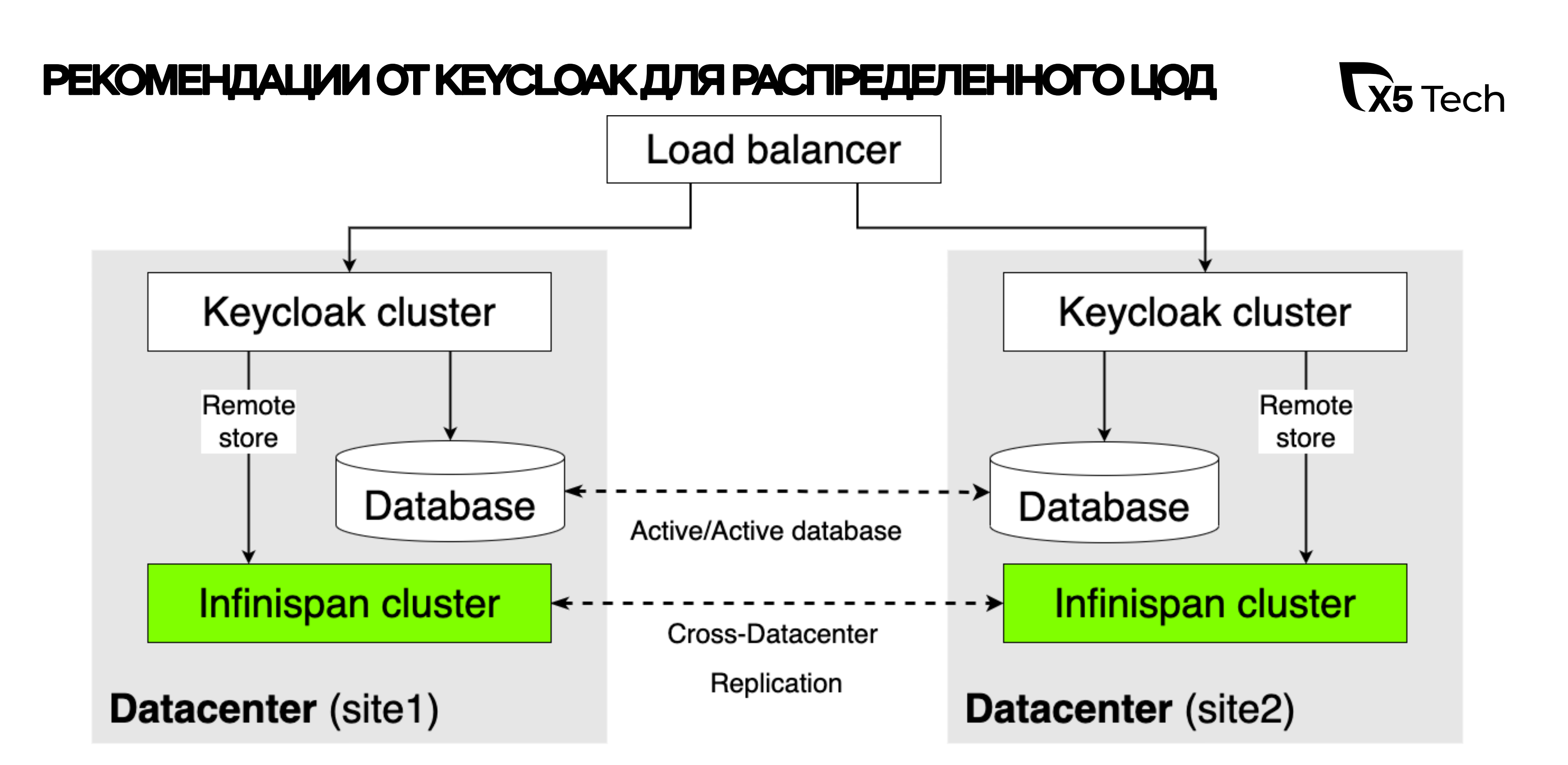
X5 Tech

TIMEOUT
TIMEOUT
TIMEOUT
Keycloak
TIMEOUT
TIMEOUT
TIMEOUT
LOADING...
INFINISPAN
POSTGRES

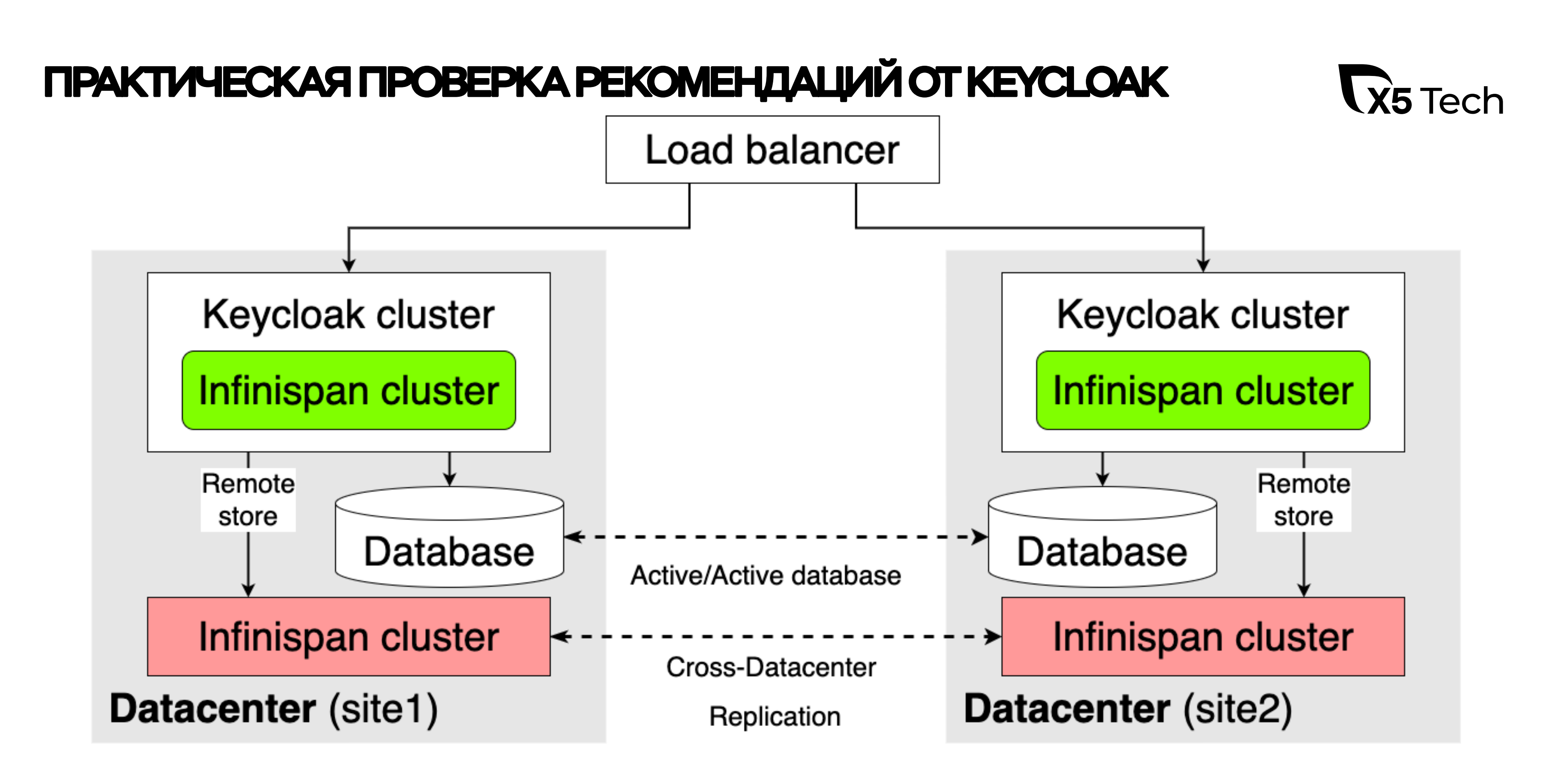
Каждый раз при загрузке происходит **прогрев кэша в Infinispan** из Postgres – перенос всех сессий из Postgres в кластер Infinispan

РЕКОМЕНДАЦИИ ОТ KEYCLOAK ДЛЯ РАСПРЕДЕЛЕННОГО ЦОД

X5 Tech



В теории Infinispan можно вынести в **отдельный кластер** от Keycloak



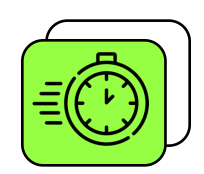
В реальности Infinispan **вынести в отдельный кластер** проблематично

ВЫВОД ПО РЕЗУЛЬТАТУ ПРОВЕРКИ РЕКОМЕНДАЦИЙ ОТ KEYCLOAK



Infinispan все же **остался внутри** Keycloak





БЫСТРОЕ РЕШЕНИЕ

КЭШИ KEYCLOAK. КАК ПРЕДСТАВЛЕНЫ В INFINISPAN



Тип (Infinispan)	Local	Replicated	Distributed
Репликация	Нет	Да (на каждой ноде)	Да
Представители	realm, client, role, user and metadata	work	loginFailures, actionTokens, *Sessions, **OfflineUserSessions**
Ограничен в размере	Да	**Нет**	**Нет**
Вайп без последствий	Да	Да	**Нет**

Данные в преведенной таблице отражают картину настройки кэшей **по умолчанию**

Настройка OfflineUserSessions





Infinispan

- cacheOwners (2)*

* В скобочках приведены значения параметров по умолчанию

Настройка OfflineUserSessions





Infinispan

- cacheOwners (2)*



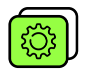
Postgres

- MAX_POOL_SIZE
- CREATE INDEX sessions_per_segment ... (created_on, offline_flag, user_session_id)

* В скобочках приведены значения параметров по умолчанию

Настройка OfflineUserSessions





Infinispan

- cacheOwners (2)*



Keycloak

- jboss.as.management.blocking.timeout (300 секунд)
- sessionsPerSegement (~500)



Postgres

- MAX_POOL_SIZE
- CREATE INDEX sessions_per_segment … (created_on, offline_flag, user_session_id)

* В скобочках приведены значения параметров по умолчанию

Настройка OfflineUserSessions





Infinispan

- cacheOwners (2)*



Keycloak

- jboss.as.management.blocking.timeout (300 секунд)
- sessionsPerSegement (~500)



Postgres

- MAX_POOL_SIZE
- CREATE INDEX sessions_per_segment … (created_on, offline_flag, user_session_id)



Kubernetes

- replicas

* В скобочках приведены значения параметров по умолчанию

НЕ РЕШАЕМ ПРОБЛЕМУ, А ОТКЛАДЫВАЕМ ЕЕ



Результат:
Добились даунтайма для 300 000 пользователей в **3 минуты!**

НЕ РЕШАЕМ ПРОБЛЕМУ, А ОТКЛАДЫВАЕМ ЕЕ



Результат:
Добились даунтайма для 300 000 пользователей в **3 минуты!**

НЕ РЕШАЕМ ПРОБЛЕМУ, А ОТКЛАДЫВАЕМ ЕЕ



Результат:
Добились даунтайма для 300 000 пользователей в **3 минуты!**

НЕ РЕШАЕМ ПРОБЛЕМУ, А ОТКЛАДЫВАЕМ ЕЕ



Время простоя, мин.

50
40
30
20
10
3
0

0,50 0,75 1,00 1,25 1,50

Количество пользователей, млн.

A
B
C

Терпимое время простоя

Результат:
Добились даунтайма для 300 000 пользователей в **3 минуты!**

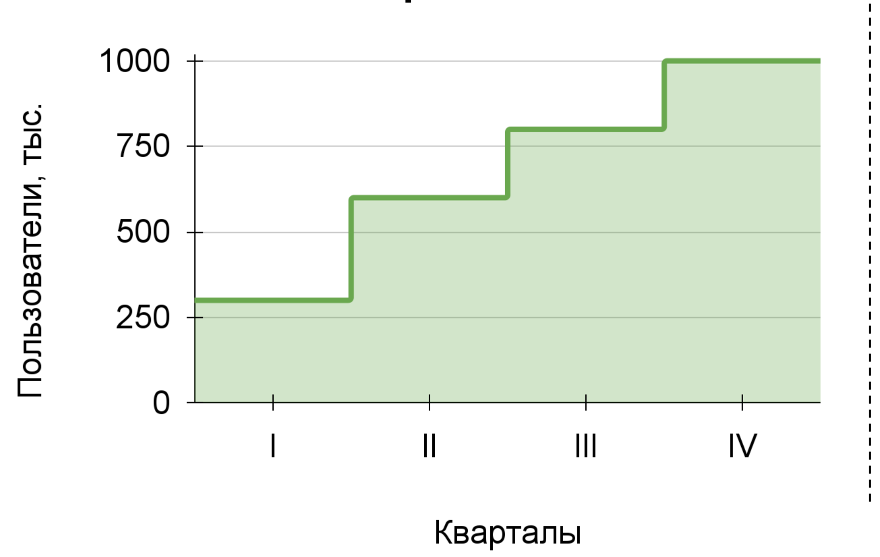


ПЕРВЫЙ
1 000 000
ПОЛЬЗОВАТЕЛЕЙ

ОЖИДАНИЕ И РЕАЛЬНОСТЬ ПО УВЕЛИЧЕНИЮ НАГРУЗКИ



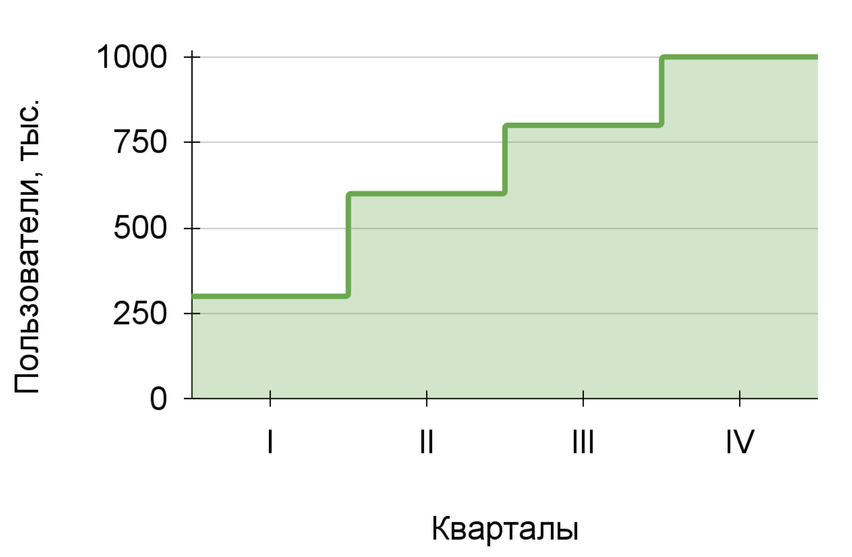
Как предполагался рост количества подключаемых к решению клиентов



ОЖИДАНИЕ И РЕАЛЬНОСТЬ ПО УВЕЛИЧЕНИЮ НАГРУЗКИ



Как предполагался рост количества подключаемых к решению клиентов



Реальный прирост клиентов



ОТЗЫВЫ ПОЛЬЗОВАТЕЛЕЙ УХУДШАЮТСЯ

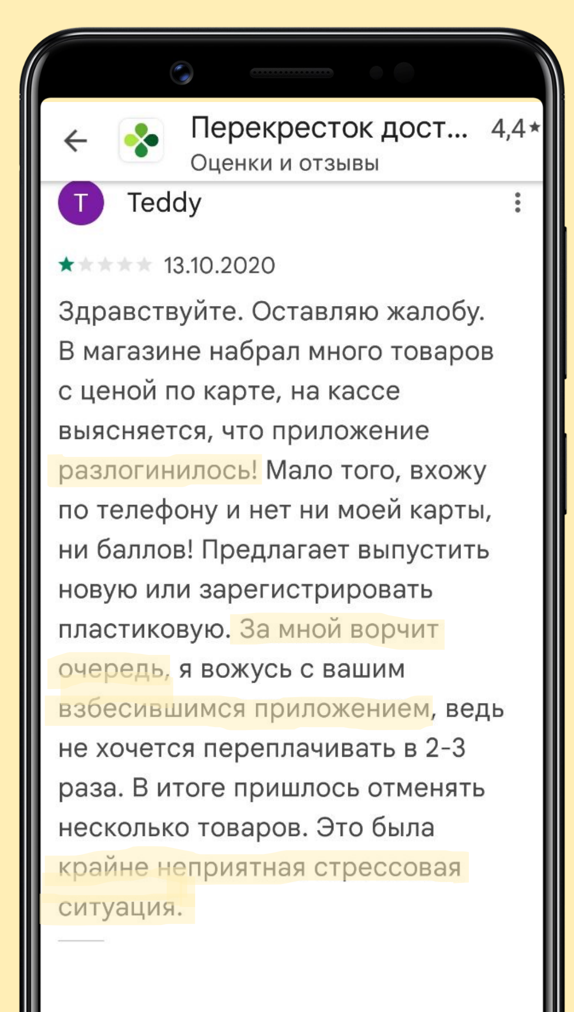
ВОРЧИТ ОЧЕРЕДЬ

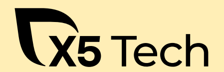
РАЗЛОГИНИЛОСЬ

ВЗБЕСИВШИЕСЯ ПРИЛОЖЕНИЕ

НЕПРИЯТНАЯ СИТУАЦИЯ

СТРЕСС





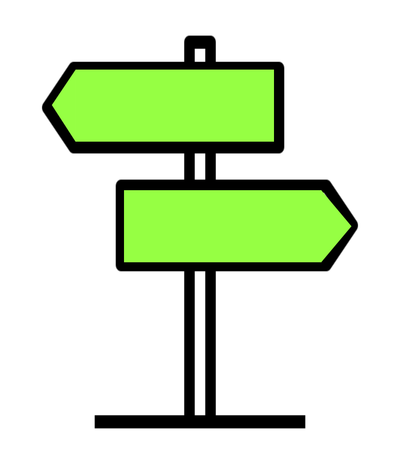
ПОИСК НАДЕЖНОГО РЕШЕНИЯ

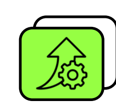




Поднятие версии

Почему нельзя просто обновиться

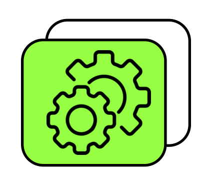




Использование подхода из более свежей версии КС (12)

Доработка 10 установленной версии в X5





ОТЧАЯННЫЙ ФОРК

ИМПЛЕМЕНТАЦИЯ PR-СООБЩЕСТВА ДЛЯ КС 12 В КС 10



KEYCLOAK-11019 Initial support for lazy offline user-session loading #7722

Closed thomasdarimont wants to merge 6 commits into keycloak:master from thomasdarimont:issue/KEYCLOAK-11019-lazy-offline-session-loading

За основу взято решение для Keycloak 12, **несовместимое** с 10-й версией

ИМПЛЕМЕНТАЦИЯ PR- СООБЩЕСТВА ДЛЯ КС 12 В КС 10





Изменение функций по получению сессий из Infinispan

- InfinispanUserSessionProvider#getUserSession(…)
- #getUserSessions(…)

ИМПЛЕМЕНТАЦИЯ PR- СООБЩЕСТВА ДЛЯ КС 12 В КС 10





Изменение функций по получению сессий из Infinispan

- InfinispanUserSessionProvider#getUserSession(…)
- #getUserSessions(…)



Изменение логики функции инициализации сессий при старте

- InfinispanUserSessionProviderFactory#loadPersistentSessions

ИМПЛЕМЕНТАЦИЯ PR- СООБЩЕСТВА ДЛЯ КС 12 В КС 10





Изменение функций по получению сессий из Infinispan

- InfinispanUserSessionProvider#getUserSession(…)
- #getUserSessions(…)



Изменение логики функции инициализации сессий при старте

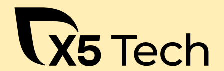
- InfinispanUserSessionProviderFactory#loadPersistentSessions



Расширение классов работы с БД

- JpaUserSessionPersisterProvider
- UserSessionPredicate

ОТ КОРОБОЧНОГО РЕШЕНИЯ В ОПАСНЫЕ ВОДЫ



Что получаем?


Условно стабильное решение с быстрым стартом

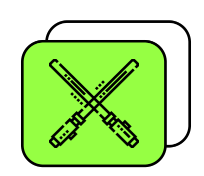

Теряем поддержку со стороны сообщества


Приобретаем свои уникальные ошибки


Прощаемся с обновлениями







ПУТЬ ДЖЕДАЯ

РЕШЕНИЕ





Повышаем версию

16+ лучше, чем в 14 версии



Покрываем решение тестами

Без своих провайдеров не обойтись



Не храним пользователей в БД Keycloak

Стратегически разделяем идентификацию
и аутентификацию в разных системах



Проектируем архитектуру, следуем стандартам

Надежные алгоритмы – гарант безопасности

Lifehack: Становитесь контрибьютором КС :)

Отказоустойчивая архитектура



ЦОД 1

ЦОД 2

Failover

Active cluster
Keycloak

Standby cluster
Keycloak

Infinispan

Postgres

Postgres

Patroni cluster

Postgres

Infinispan

HAPROXY + +

etcd

ВЫВОДЫ





SSO – это удобно



Писать с нуля сервер аутентификации долго и сложно



Техническую документацию пишут люди. Доверяй, но проверяй



Эксперименты – хорошо, но в меру, когда речь идет о решении на многомиллионную аудиторию

Обратная связь и комментарии по докладу по ссылке



Контакты Telegram:

Ирина Блажина @A_Blair

Николай Зайцев @gtjbtits

